

Abstract

A slice plane, oriented parallel to a viewing plane, is passed through a cuboidal dataset at regular intervals. The intersection of the slice plane with the cuboidal volume dataset results in primitives (quads, triangles, etc. depending on the angle and position of the intersection) whose vertices have position coordinates (x_u , y_u , z_u) and 3D-texture coordinates (r , s , t). The resulting primitives are rasterized using, for example, a traditional 3D graphics pipeline wherein the 3D-texture coordinates are interpolated across the scanlines producing 3D-texture coordinates for each fragment. The resulting 3D-texture coordinates for each fragment are stored in a 2D-texture storage area. These 2D-textures are called density-textures. By preprocessing the cuboidal dataset, the rendering process becomes a compositing process. A rendering process is comprised of looking-up, for each densel in the texture, the corresponding color and opacity values in the current lookup-table. A user-specified compositing function is used to blend the values with those in the framebuffer to arrive at the final result. The final result, i.e. the values in the framebuffer, is displayed.